Model 4: KNN with SMOTE Across All Groups[34]

- **Objective**: To classify individuals based on lifestyle factors and predict diabetes development.
- **Method**: KNN using SMOTE across all age groups.
- **Results**: Using KNN and SMOTE to help balance the data and review results using ROC curve to determine the performance of a classification model on the data across all ages using the same lifestyle variables as features.
    - ['Smoker', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholConsump']
    - 70/30 split
    - Standardized

Data Preparation

The first classification model to be performed on the dataset is k-nearest neighbors. SMOTE was also used for this portion as to help with the imbalance dataset, and to avoid overfitting the model. The first KNN model will be created across all age groups within the dataset. This model used a 70/30 training testing split.

Model Training and Evaluation

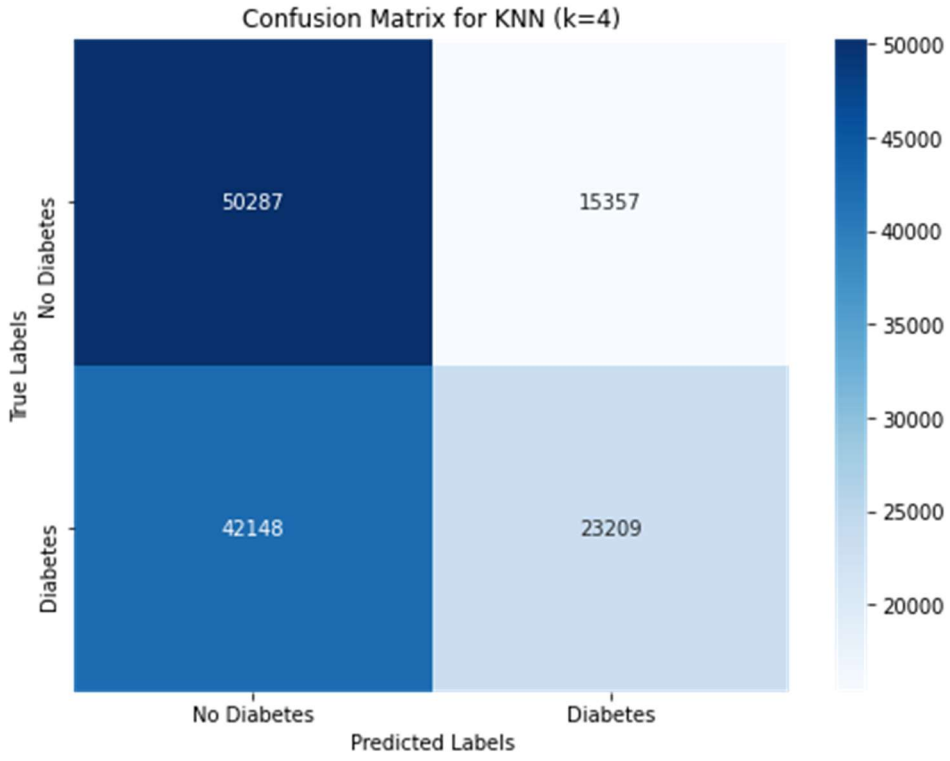
The first step in building the KNN model was to find the most optimal K-value. A "for loop" was created using "Accuracy" as the performance metric in order to determine the proper k-value to use for modeling. The chart below shows that a K-value of 4 was the best option for analysis.

K Value	Accuracy
1	0.487042
2	0.506324
3	0.546087
4	0.561034
5	0.557339

[34] Source code here

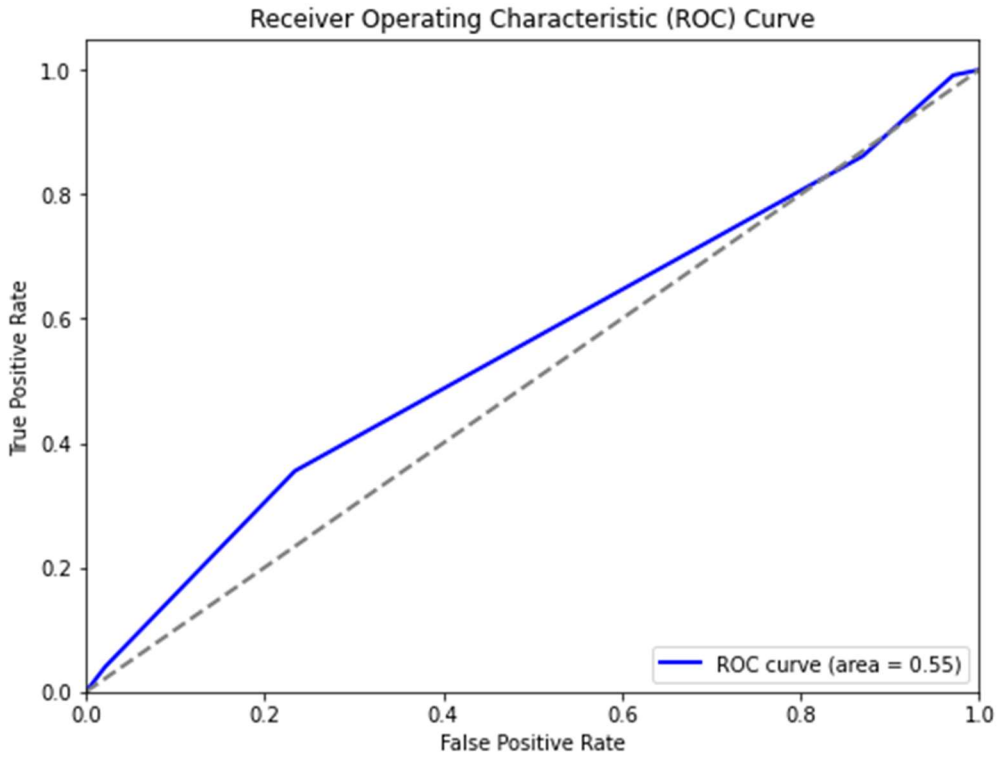
The above table shows the accuracy ratings for values of k ranging 1-5. From this analysis, it is determined that a k-value of 4 nearest neighbors provides the best result for the ensuing KNN models.

Confusion Matrix for Model Evaluation



- The confusion matrix metrics shown in the classification report show a notable discrepancy in performance between non-diabetic and diabetic classifications. While it has a moderate ability to correctly identify non-diabetics, it performs poorly in identifying diabetics.

ROC Curve for Model Evaluation



- The ROC curve shown for the KNN model across all age groups. AUC value was calculated at 0.55, suggesting the model does not have a strong predictive ability.

Model 4: SMOTE KNN Classification Report				
	Precision	**Recall**	**F1-Score**	**Support**
0.0	0.54	0.77	0.64	65644
1.0	0.60	0.36	0.45	65357
Accuracy			0.56	131001
Macro Avg	0.57	0.56	0.54	131001
Weighted Avg	0.57	0.56	0.54	131001
ROC AUC SCORE		**0.5518912009020336**		

The current model with an AUC of 0.55 indicates that lifestyle factors such as smoking, physical activity, fruit and vegetable consumption, and heavy alcohol consumption have limited predictive capability for diabetes status, though the AUC is indicative that the model has a better predictive capability than random guessing. Improving model performance will likely require a thorough investigation of the underlying data patterns, as well as potentially looking at certain subclasses in the data, which will be performed as we create a more advanced model by breaking up the data into age groups.

Model 5: KNN with SMOTE for Individual Age Groups[35]

- **Objective**: To determine the lifestyle factors that are most strongly associated with the development of diabetes in different age groups.
- **Method**: KNN using Synthetic Minority Sampling (SMOTE) for each subgroup.
- **Description:** This was the second model which used all the lifestyle factors for the whole dataset. It uses SMOTE to attempt to balance out the data set and achieve better AUC. It was created with these features of interest
    - ['Smoker', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholConsump']
    - 70/30 split

[35] Source code here

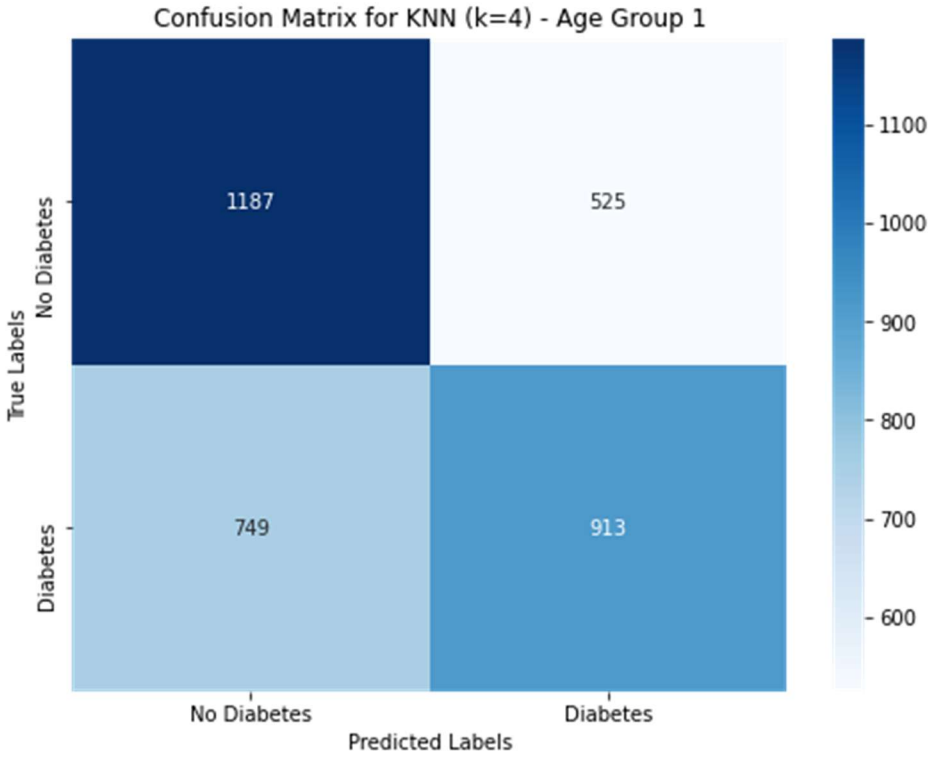
- Standardized
- k-Value = 4
- Created for each individual age subgroup

Data Preparation

The next step to continue modeling for RQ1 is to complete a classification task for the data. This will be done using the k-Nearest Neighbors method for each of the 13 individual age groups. In order to stay consistent with the regression tasks performed so far, we will use a 70/30 training testing split, and recruit the help of SMOTE to help with the imbalanced data.

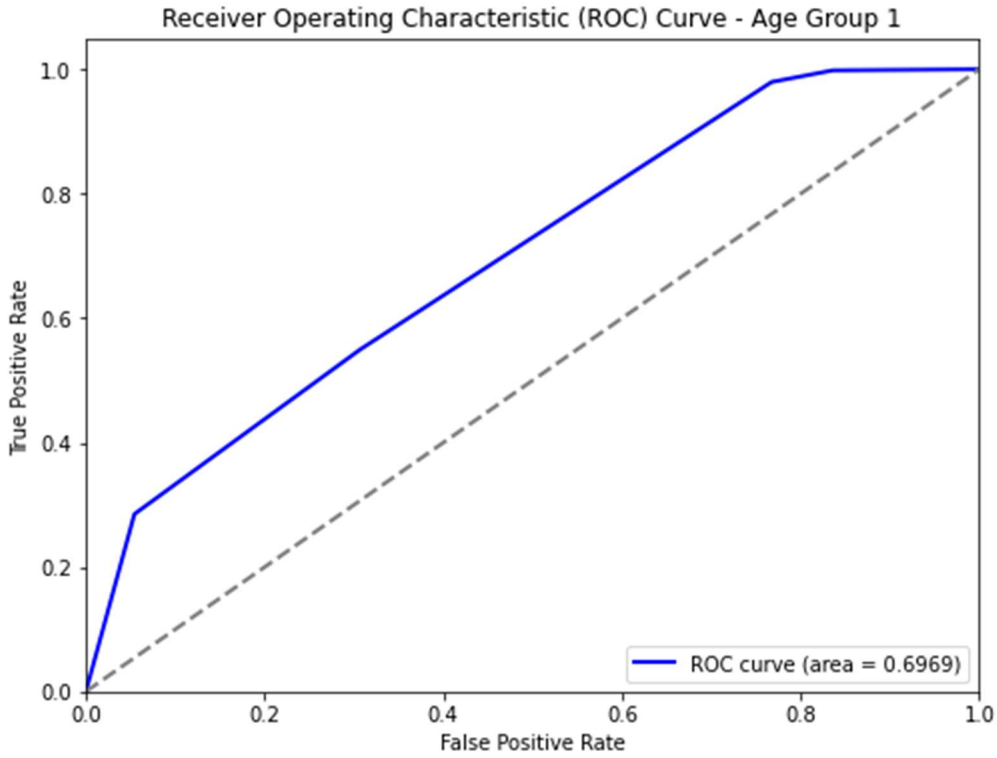
Model Training and Evaluation

- Confusion Matrix for Age Group 1



- The confusion matrix for KNN on Age Group 1 indicates that the predictive ability is much more accurate than for the model across all age groups.

- ROC Curve for Age Group 1



- The ROC curve shown has an AUC value of 0.6969, indicating relatively strong predictive power when the model is ran for Age Group 1.

- Classification Report for all 13 Age Groups

Age Group	Accuracy		Precision	Recall	F1-Score	ROC AUC Score
1: 18-24	62%	**Non-Diabetes**	0.61	0.69	0.65	**0.6969**
		Diabetes	0.63	0.55	0.59	
2: 25-29	60%	**Non-Diabetes**	0.57	0.78	0.66	**0.6260**
		Diabetes	0.67	0.44	0.53	
3: 30-34	53%	**Non-Diabetes**	0.52	0.77	0.62	**0.5358**
		Diabetes	0.55	0.28	0.37	
4. 35-39	53%	**Non-Diabetes**	0.52	0.93	0.67	**0.5773**
		Diabetes	0.66	0.14	0.23	
5. 40-44	51%	**Non-Diabetes**	0.51	0.48	0.50	**0.5197**
		Diabetes	0.50	0.53	0.52	
6. 45-49	58%	**Non-Diabetes**	0.56	0.66	0.61	**0.6166**
		Diabetes	0.59	0.49	0.54	
7. 50-54	55%	**Non-Diabetes**	0.53	0.81	0.64	**0.5604**
		Diabetes	0.60	0.28	0.39	
8. 55-59	51%	**Non-Diabetes**	0.51	0.72	0.59	**0.5279**
		Diabetes	0.51	0.30	0.38	
9. 60-64		**Non-Diabetes**	0.54	0.70	0.61	

	55%	**Diabetes**	0.57	0.40	0.47	**0.5796**
10. 65-69	52%	**Non-Diabetes**	0.51	0.92	0.66	**0.5405**
		Diabetes	0.62	0.13	0.21	
11. 70-74	52%	**Non-Diabetes**	0.51	0.86	0.64	**0.5300**
		Diabetes	0.57	0.18	0.27	
12. 75-79	56%	**Non-Diabetes**	0.56	0.69	0.62	**0.5657**
		Diabetes	0.57	0.43	0.49	
13. 80 or older	53%	**Non-Diabetes**	0.52	0.78	0.62	**0.5257**
		Diabetes	0.56	0.28	0.37	

The KNN model broken up by age group provided much more valuable insights to the question being answered. Much like the first model, a confusion matrix and ROC curve was created for each individual age group. The visualizations for Age Group 1 (18-24 years old) were chosen to showcase in this analysis as that group performed best in this model. Age Group 2 (25-29 years old) follows close behind the first Group with 60% Accuracy and an AUC score of 0.6260.

Based on these results from the first two age groups, the model does suggest that in young adulthood (18-30 years of age) lifestyle factors are relatively important to the development of diabetes. To expand on that, results listed in the classification report suggest that as age increases, lifestyle factors alone are not necessarily the most important variables in predicting diabetes diagnosis. This could be due in part to how the data was collected and the definitions of each feature. A dataset that provides more in depth data such as how often someone exercises at a moderate to intense level rather than just if they have exercised in the last 30 days could provide more meaningful information in reviewing the impact of lifestyle factors in older ages.

Model 4: KNN with SMOTE Across All Groups

```
from ucimlrepo import fetch_ucirepo

cdc_diabetes_health_indicators = fetch_ucirepo(id=891)

X = cdc_diabetes_health_indicators.data.features
y = cdc_diabetes_health_indicators.data.targets

print(cdc_diabetes_health_indicators.metadata)

print(cdc_diabetes_health_indicators.variables)

X_lifestyle = X[['Smoker', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholConsump']]

print(y.head())
print(X_lifestyle.head())

from imblearn.over_sampling import SMOTE

smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_lifestyle, y)

from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_standardized = scaler.fit_transform(X_resampled)

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X_standardized, y_resampled, test_size = 0.30,
random_state = 0)
```

```
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from  sklearn.metrics import accuracy_score
```

```
results = []
for k in range(1,10):
  knn = KNeighborsClassifier(n_neighbors=k).fit(X_train, y_train)
  y_pred = knn.predict(X_test)
  results.append({
      "k":k,
      "accuracy":accuracy_score(y_test, y_pred)
  })

results = pd.DataFrame(results)
results

knn_4 = KNeighborsClassifier(n_neighbors=4)
knn_4.fit(X_train, y_train)
y_pred_4 = knn_4.predict(X_test)
```

Visualize Confusion Matrix and ROC Curve

```
from sklearn.metrics import confusion_matrix
knn_cm = confusion_matrix(y_test, y_pred_4)

import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(8, 6))
sns.heatmap(knn_cm, annot=True, fmt='d', cmap='Blues', xticklabels=['No Diabetes', 'Diabetes'],
yticklabels=['No Diabetes', 'Diabetes'])
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix for KNN (k=4)')
plt.show()

from sklearn.metrics import roc_curve, roc_auc_score

y_pred_prob = knn_4.predict_proba(X_test)[:, 1]

fpr, tpr, thresholds = roc_curve(y_test, y_pred_prob)
auc = roc_auc_score(y_test, y_pred_prob)
```

```
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, color='blue', lw=2, label=f'ROC curve (area = {auc:.2f})')
plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()

print(f'AUC: {auc}')
```

Classification Report for Evaluation Metrics

```
from sklearn.metrics import classification_report

report = classification_report(y_test, y_pred_4, target_names=['No Diabetes', 'Diabetes'],
output_dict=True)

report_df = pd.DataFrame(report).transpose()

print(report_df)
```

Model 5: KNN with SMOTE for Individual Age Groups

```
import pandas as pd
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix, roc_curve, roc_auc_score, classification_report,
accuracy_score
import seaborn as sns
import matplotlib.pyplot as plt

cdc_diabetes_health_indicators =
pd.read_csv('diabetes_binary_health_indicators_BRFSS2015.csv')

def analyze_age_group(age_group):
    age_group_data = cdc_diabetes_health_indicators[cdc_diabetes_health_indicators['Age'] ==
age_group]

    y = age_group_data['Diabetes_binary']
    X = age_group_data[['Smoker', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholConsump']]

    smote = SMOTE(random_state=42)
    X_resampled, y_resampled = smote.fit_resample(X, y)

    scaler = StandardScaler()
    X_resampled_scaled = scaler.fit_transform(X_resampled)

    X_train, X_test, y_train, y_test = train_test_split(X_resampled_scaled, y_resampled,
test_size=0.3, random_state=42)

    knn_4 = KNeighborsClassifier(n_neighbors=4)
    knn_4.fit(X_train, y_train)

    y_pred_4 = knn_4.predict(X_test)

    knn_cm = confusion_matrix(y_test, y_pred_4)

    plt.figure(figsize=(8, 6))
    sns.heatmap(knn_cm, annot=True, fmt='d', cmap='Blues', xticklabels=['No Diabetes',
'Diabetes'], yticklabels=['No Diabetes', 'Diabetes'])
```

```
    plt.xlabel('Predicted Labels')
    plt.ylabel('True Labels')
    plt.title(f'Confusion Matrix for KNN (k=4) - Age Group {age_group}')
    plt.show()

    y_pred_prob = knn_4.predict_proba(X_test)[:, 1]

    fpr, tpr, thresholds = roc_curve(y_test, y_pred_prob)
    auc = roc_auc_score(y_test, y_pred_prob)

    plt.figure(figsize=(8, 6))
    plt.plot(fpr, tpr, color='blue', lw=2, label=f'ROC curve (area = {auc:.4f})')
    plt.plot([0, 1], [0, 1], color='gray', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Receiver Operating Characteristic (ROC) Curve - Age Group {age_group}')
    plt.legend(loc="lower right")
    plt.show()

    print(f'AUC for Age Group {age_group}: {auc:.4f}')

    report = classification_report(y_test, y_pred_4, target_names=['No Diabetes', 'Diabetes'],
output_dict=True)

    report_df = pd.DataFrame(report).transpose()

    print(f'Classification Report for Age Group {age_group}:')
    print(report_df)

    acc = accuracy_score(y_test, y_pred_4)
    print(f'Accuracy: {acc:.2f}')

for age_group in range(1, 14):
    analyze_age_group(age_group)
```